

**UNIVERZITA KARLOVA V PRAZE**

**Přírodovědecká fakulta**

Katedra aplikované geoinformatiky a kartografie



**MAPOVÝ SERVER PRO VIZUALIZACI  
KAMPUSU ALBERTOV**

**MAP SERVER VISUALIZING  
THE ALBERTOV CAMPUS**

Bakalářská práce

Petr Polášek

květen 2012

Vedoucí bakalářské práce: RNDr. Přemysl Štych, Ph.D.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a že jsem všechny použité prameny řádně citoval.

Jsem si vědom toho, že případné použití výsledků, získaných v této práci, mimo Univerzitu Karlovu v Praze je možné pouze po písemném souhlasu této univerzity.

Svoluji k zapůjčení této práce pro studijní účely a souhlasím s tím, aby byla řádně vedena v evidenci vypůjčovateli.

V Praze dne 28. května 2012

.....  
Petr Polášek

## **Poděkování**

Na tomto místě bych rád poděkoval vedoucímu mé práce RNDr. Přemyslu Štychovi, Ph.D. za jeho cenné připomínky a poskytnuté rady. Dále bych chtěl poděkovat svým rodičům za podporu během celého studia.

## **Mapový server pro vizualizaci kampusu Albertov**

### **Abstrakt**

Základním cílem této práce je vytvořit interaktivní webovou aplikaci umožňující prohlížení několika mapových vrstev kampusu Albertov a okolí. Aplikace bude poskytovat informace o univerzitních budovách a bude zajištěna pomocí Open Source nástrojů MapServer a OpenLayers. Rešeršní část práce se zabývá možnostmi zpřístupnění digitálních map na internetu, webovými mapovými službami a architekturou mapových serverů. V praktické části jsou popsány oba dva nástroje MapServer a OpenLayers a je zde popsána metodika práce.

**Klíčová slova:** MapServer, OpenLayers, mapový server

## **Map server visualizing the Albertov campus**

### **Abstract**

The basic aim of this work is to create an interactive web application which enables to display several map layers of the Albertov campus and its surroundings. The application will be providing information about university buildings and it will be created with Open Source tools MapServer and OpenLayers. In the first part the work deals with digital internet maps, web map services and with architecture of map servers. In the practical part are described both MapServer and OpenLayers and also the methodology of the work.

**Keywords:** MapServer, OpenLayers, map server

## OBSAH

<b>Přehled použitých zkratk</b> .....	<b>8</b>
<b>Seznam obrázků</b> .....	<b>10</b>
<b>1 Úvod</b> .....	<b>11</b>
<b>2 Uvedení do problematiky</b> .....	<b>13</b>
2.1 Digitální mapa .....	13
2.1.1 Výhody digitálních map .....	14
2.2 Internetové mapy a GIS .....	14
2.2.1 Typy internetových map .....	14
2.2.2 Funkce GIS .....	15
2.2.3 Typy architektury GIS .....	15
2.3 Klientské aplikace .....	16
2.4 Interoperabilita a standardy poskytování dat .....	17
2.4.1 Základní OGC standardy pro sdílení geografických dat .....	17
2.4.2 EPSG kódy a souřadnicové systémy .....	20
<b>3 Mapové servery</b> .....	<b>21</b>
3.1 Komponenty mapového serveru .....	21
3.2 Používané technologie mapových serverů .....	23
3.3 Web Mapping API .....	24
<b>4 Použitá data a metodika práce</b> .....	<b>25</b>
4.1 Struktura MapServeru .....	25
4.2 Zprovoznění MapServeru .....	29
4.3 Tvorba webového rozhraní pomocí API OpenLayers .....	30
4.3.1 JavaScript .....	30
4.3.2 Struktura OpenLayers .....	31
4.4 Mapový server kampusu Albertov .....	33
4.4.1 Použité mapové vrstvy .....	33
4.4.2 MapServer jako WMS služba .....	34
4.4.3 Nastavení souřadnicového systému .....	35

4.4.4 Ovládací prvky webového rozhraní OpenLayers .....	35
<b>5 Výsledky.....</b>	<b>37</b>
<b>6 Diskuse a závěr .....</b>	<b>38</b>
<b>Seznam zdrojů a informací.....</b>	<b>39</b>
<b>Seznam příloh .....</b>	<b>41</b>

## PŘEHLED POUŽITÝCH ZKRATEK

<b>AJAX</b>	Asynchronous JavaScript and XML
<b>API</b>	Application Programming Interface
<b>CGI</b>	Common Gateway Interface
<b>CSS</b>	Cascading Style Sheets
<b>CT</b>	Coordinate Transformation Service
<b>EPSG</b>	EPSG Geodetic Parameter Dataset
<b>ESRI</b>	Environmental Systems Research Institute
<b>GDAL</b>	Geospatial Data Abstraction Library
<b>GIF</b>	Graphics Interchange Format
<b>GIS</b>	Geografický Informační Systém
<b>GML</b>	Geography Markup Language
<b>GPS</b>	Global Positioning System
<b>HTML</b>	HyperText Markup Language
<b>HTTP</b>	Hypertext Transfer Protocol
<b>KML</b>	Keyhole Markup Language
<b>MS4W</b>	MapServer for Windows
<b>NASA</b>	National Aeronautics and Space Administration
<b>OGC</b>	Open Geospatial Consortium
<b>OGR</b>	OGR Simple Features Library
<b>OSM</b>	OpenStreetMap
<b>PHP</b>	PHP: Hypertext Preprocessor
<b>S-JTSK</b>	Systém Jednotné Trigonometrické Sítě Katastrální
<b>SQL</b>	Structured Query Language

---

<b>TIFF</b>	Tag Image File Format
<b>UMN</b>	University of Minnesota
<b>URL</b>	Uniform Resource Locator
<b>URM</b>	Útvar rozvoje hlavního města Prahy
<b>WCS</b>	Web Coverage Service
<b>WFS</b>	Web Feature Service
<b>WGS 84</b>	World Geodetic System 1984
<b>WMS</b>	Web Map Service
<b>W3C</b>	World Wide Web Consortium
<b>XML</b>	eXtensible Markup Language



## SEZNAM OBRÁZKŮ A TABULEK

Tab. 1	Rozdíl mezi tenkým a tlustým klientem.....	17
Tab. 2	Významné souřadnicové systémy a jejich parametry .....	20
Obr. 1	Schéma mapového serveru .....	22
Obr. 2	Struktura MapServeru .....	26
Obr. 3	Ukázka Map File.....	27
Obr. 4	Ukázka webového rozhraní definovaného šablonou „itasca_basic.html“ .....	29
Obr. 5	Schéma klientského skriptu .....	30
Obr. 6	Ukázka kódu, který do webového rozhraní OpenLayers přidá vrstvu OSM .....	32
Obr. 7	Základní webové rozhraní OpenLayers s vrstvou OSM.....	33

# KAPITOLA 1

## Úvod

Dnešní svět je doslova protkán informačními technologiemi, mezi kterými dominuje internet. Díky tomu se svět zdánlivě zmenšuje a není problém komunikovat s lidmi od nás vzdálených na tisíce kilometrů nebo sledovat dění na druhé straně planety. To nic nemění na tom, že se stále pohybujeme ve světě reálném, kde nám k orientaci slouží mapy. Dnes, kdy máme přístup k GPS, velkokapacitním diskům a především webovým technologiím s připojením k internetu, nemusíme sahat po klasických papírových mapách, ale můžeme plně využívat výhody digitálních map v kombinaci s rozvíjejícími se geoinformačními technologiemi. Současně s tím se také vyvíjí fenomén Open Source, a tak může téměř každý nejen geoinformačními technologiemi využívat, ale také se podílet na vytváření nových informačních zdrojích, bez nutnosti investovat do drahých softwarových produktů.

Hlavním smyslem této práce bylo vytvoření mapového serveru kampusu Albertov a okolí, který by měl sloužit pro lepší orientaci ve změní budov na Albertově. Uživatel by měl mít možnost získat základní informace o univerzitních budovách a neměla chybět volba několika podkladových vrstev. Mapový server by měl využívat pouze Open Source technologie.

Práce má dvě stěžejní části - rešeršní a praktickou. V rešeršní části je krátce pojednáno o digitálních mapách a základních možnostech jejich šíření na internetu. Protože se obecně tato oblast GIS neustále vyvíjí, objevují se nové formáty pro popis a sdílení geodat, bylo třeba zavést standardy zajišťující interoperabilitu mezi jednotlivými nástroji. Tímto tématem se zabývá kapitola Interoperabilita a standardy poskytování dat, kde je kromě základních standardů pro sdílení geodat nastíněna problematika souřadnicových systémů. Následuje kapitola o mapových serverech, které jsou nepostradatelnou součástí pro šíření a vizualizaci prostorových dat na internetu. Je zde pojednáno o základní struktuře mapových serverů a jsou zde zmíněny konkrétní nástroje pro vytvoření mapového serveru.

V metodické části jsou podrobněji popsány nástroje, které byly použity pro tvorbu mapového serveru kampusu Albertov - UMN MapServer a OpenLayers. Je zde zmíněna problematika

konfigurace mapových vrstev pro MapServer a možnosti jejich umístění do webového rozhraní, vytváření ovládacích prvků pomocí OpenLayers, přístup ke vzdáleným vrstvám WMS a další.

Tato práce si neklade za cíl dopodrobna seznámit čtenáře se všemi aspekty webových služeb týkajících se geoinformačních technologií. Taková práce by musela být nesrovnatelně obsáhlejší. Tuto práci je třeba považovat za jakýsi úvod do světa webových technologií, které umožňují práci s geografickými daty a jejich sdílení po internetu.

## KAPITOLA 2

### Uvedení do problematiky

Na samém začátku práce bylo třeba vyhledat vhodné zdroje informací, zabývající se řešenou problematikou. Jak se dalo očekávat, většina materiálů byla dostupná v elektronické podobě, ať už jako akademické práce, odborné články, či dokumentace daných nástrojů. Mezi hlavní zdroje patřily dokumentace MapServeru a OpenLayers, webové stránky OGC a práce studentů z ČVUT či PřF UK. Dalšími zdroji byly webové stránky popisovaných nástrojů, které poskytly základní informace.

#### 2.1 Digitální mapa

Mapa je mocným nástrojem, který člověk využívá po tisíce let. S mapou se setkáváme v podstatě každý den a lze ji využít při nejrůznějších činnostech. Mapa je dokonce o několik tisíc let starší než písmo. První mapy měly podobu plánek na zdech skal, nebo na hliněných a dřevěných tabulkách. Postupem času se mapy zdokonalovaly, jejich nosičem se stal papír, a tak se mohly bez větších problémů rozšířit. Na konci 20. století s rozvojem počítačů a informačních technologií vznikl nový fenomén - digitální mapy. V oblasti kartografie a mapování se tak otevřely nové možnosti a digitální mapy se staly nepostradatelnou součástí dnešního života.

„Digitální mapa je záznam obsahu a konstrukčních (případně jiných) prvků mapy, které je možno vizualizovat a zpracovávat pomocí počítačového systému“ (Čerba 2008, s. 5). Jinými slovy se jedná o prostorové informace uložené ve formě digitálních dat. Vizualizací digitální mapy pak vzniká mapa elektronická. Digitální mapy se mimo jiné uplatňují pro GPS navigace, mapové servery nebo pro různé druhy prostorových analýz. Vytváření a vizualizování digitálních map se však neobejde bez patřičných znalostí a nástrojů. Použitím nevhodných či nepřesných dat, nebo špatným zpracováním těchto dat, může vzniknout na pohled sice hezká mapa, ale s nízkou vypovídající hodnotou. Digitální mapa tedy nemusí být vždy kvalitní, a na to je třeba dát pozor. Potenciál digitálních map spočívá mimo jiné v možnosti jejich snadného zpřístupnění veřejnosti. V dnešní době má téměř každý přístup k internetu, a tak se na řadu dostávají mapové servery umožňující digitální mapy prohlížet a pracovat s nimi téměř odkudkoli.

### 2.1.1 Výhody digitálních map

Digitální mapování má nesporné výhody oproti mapování klasickému. Změní-li se např. velikost nebo poloha nějakého objektu zobrazeného na klasické mapě, je třeba vytvořit mapu novou. Stejně tak v případě mapy rozrůstajícího se města, která po několika letech bude zahrnovat jen jeho část, a bude tak třeba vytvořit kompletně novou mapu v jiném měřítku. Tyto problémy jsou odstraněny digitálním mapováním kde jsou veškerá geodata uložena v databázi, a z nich jsou vytvářeny mapové vrstvy. Změní-li se parametry nějakého objektu, stačí upravit pouze daný objekt a nemusí se vytvářet nová mapa. Problém s měřítkem u digitálního mapování zcela odpadá, protože digitální prostorová data měřítko nemají. Měřítko má až jejich obrazová reprezentace a to se dá libovolně měnit, bez žádných zásahů do mapy. Navíc změny provedené u zdrojových dat digitální mapy jsou okamžitě reflektovány v jejich obrazové reprezentaci. Digitální mapy by tak měly zobrazovat aktuální stav skutečnosti, na rozdíl od map klasických, které jsou už v době tisku „zastaralé“. (Mitchell 2005)

## 2.2 Internetové mapy a GIS

Efektivní způsob zpřístupnění digitálních map široké veřejnosti spočívá ve využití internetu v kombinaci s vhodným geoinformačním systémem (dále GIS). GIS může mít různé podoby a může plnit různé funkce, ale základ všech GIS je stejný.

### 2.2.1 Typy internetových map

Na internetu lze prezentovat mapy dvěma základními způsoby: statickou nebo interaktivní formou.

#### Statická mapa

Statická mapa není nic jiného než obyčejný obrázek. Může se jednat o klasickou naskenovanou mapu nebo o mapu vykreslenou GIS softwarem. K publikování statické mapy stačí vytvořit jednoduchou webovou stránku, na kterou se mapa umístí v podobě obrázku. V těchto mapách není možnost změny měřítka, posunu či prostorového dotazu.

#### Interaktivní mapa

Interaktivní mapy jsou mnohem zajímavějším řešením než mapy statické. Z názvu je zřejmé, že uživatel může s mapou nějak interagovat - ovládat ji. Uživatel si může vybrat jaké mapové vrstvy chce zapnout/vypnout, může přiblížit mapu na konkrétní území, může zjistit vlastnosti objektů na mapě atd. Tvůrce interaktivní mapy tedy nemusí přemýšlet, co konkrétně by se chtěl uživatel z mapy dozvědět, tuto volbu nechá na uživateli samotném. Příkladem takové mapy může být portál Mapy.cz nebo Google Maps. Interaktivní mapy jsou velmi užitečné, ale vytvoření takové mapy je mnohem komplikovanější než vytvoření mapy statické. V podstatě se jedná o GIS systém využívající internet, skládající se z webového serveru, který komunikuje s mapovým serverem

generujícím z datových zdrojů na základě požadavků klienta výslednou mapu (více v kapitole 3 - Mapové servery).

### 2.2.2 Funkce GIS

GIS je nepostradatelným nástrojem při práci s prostorovými daty a tedy i při vizualizaci digitálních map. Může pak vzniknout dojem, že hlavní funkcí GIS je právě vytváření map. Mapa je ale výsledkem mnoha procesů, které GIS vykonává. Jedná se hlavně o analytický nástroj, pracující s prostorovými vztahy mezi objekty. Mapy pak plní funkci zobrazení informací (Minář 2008). V mnoha případech se výsledek skládá pouze z tabulkových dat, které přiřazují kvantitativní informace určitému místu na Zemi.

GIS může být definován různě. Každý GIS ale musí mít některé základní schopnosti:

- **Manipulace s daty** - zahrnuje nástroje pro práci s databázemi, které uchovávají prostorová data, schopnost importovat a převést data z různých zdrojů do podoby vhodné pro další zpracování, spravovat atributy a prostorové informace uložených dat (data mohou obsahovat chyby, které je třeba odstranit), možnost převodu mezi různými modely reprezentace dat (vektorizace, rasterizace).
- **Analýza dat** - hlavní schopnost GIS, umožňuje ptát se na prostorový vztah elementů dat (elementy jsou poloha prvků a jejich atributy), např. kde se nachází určitý typ prvků, nebo jak se změnilo rozložení daného prvku v čase. Mimo to je možné provádět „neprostorové“ dotazy, např. kolik jezer má větší plochu než 10 m<sup>2</sup>.
- **Georeferencování** - proces přiřazování prvkům zeměpisné souřadnice.
- **Prezentace výsledků** - vytvoření požadovaných výstupů ze zpracovaných dat. Může jít jak o grafický výstup v podobě mapy, tak o výstup v podobě tabulek.

(Kropla 2005)

### 2.2.3 Typy architektury GIS

V průběhu vývoje GIS došlo k několika změnám v jejich architektuře. První GIS vznikl koncem 60. let 20. století, kdy se jednalo o tzv. „mainframe“ architekturu, využívající sálový počítač. Postupem času se začaly využívat desktopové GIS a v současnosti se přechází k distribuovaným GIS.

- **Tradiční desktopové GIS** - označuje se také jako „stand - alone“ GIS. Všechna data, programy a uživatelské rozhraní jsou umístěny na jednom osobním počítači.

- **Klient - server GIS** - dvouvrstvý model, kdy desktop GIS (klient) má možnost komunikovat se vzdáleným serverem. Stejně jako tradiční desktop GIS je i tento model závislý na platformě.
- **Distribuované GIS** - jsou založeny na architektuře distribuovaných komponent, které mezi sebou komunikují prostřednictvím internetu nezávisle na platformách, operačních systémech či použitých programovacích jazycích. Nerozlišuje se zde mezi klientem a serverem, protože daný GIS může být jak klient, tak server zároveň. Podmínkou ale je vysoký stupeň interoperability (o interoperabilitě více v kapitole 3.4).

(Sklenička 2006a)

## 2.3 Klientské aplikace

Při práci na internetu se využívá možnosti sdílení dat mezi vzdálenými počítači připojených k internetu. Jednotlivé počítače se pak rozlišují na klienty a servery podle toho, jestli jsou poskytovateli nebo příjemci dat. Obecně je klient žadatelem a server poskytovatelem dat nebo služeb. Klienty a servery nemusí být nutně samostatné počítače, ale může se jednat pouze o aplikaci či program. Typicky při prohlížení internetových stránek je klientem webový prohlížeč, který požaduje data nebo nějakou službu ze vzdáleného webového serveru. Klientské aplikace se potom dělí na „tenké“ a „tlusté“.

### Tenký klient

Tenký klient využívá zdroje vzdáleného serveru. Klientská aplikace pouze předá serveru příkazy ke zpracování, server tyto příkazy vykoná a klient zobrazí výsledek v grafické podobě. V případě prohlížení interaktivní mapy je na straně uživatele vyžadován webový prohlížeč podporující AJAX technologie<sup>1</sup>. U tenkého klienta je kladen důraz na jednoduché ovládání, bezpečnost dat a rychlost přístupu k datům. Tento způsob práce se zdrojovými daty méně zatěžuje počítač uživatele, ale vyžaduje vyšší technické nároky na vybavení serveru (Říha 2007).

### Tlustý klient

Tlustý klient provádí veškeré operace sám a server má význam pouze jako poskytovatel dat. V našem případě práce s geografickými daty musí klient disponovat desktopovou GIS aplikací schopnou tato data zobrazit a editovat. Architektura tlustého klienta klade vyšší nároky na uživatelův počítač a méně zatěžuje server (Říha 2007).

---

<sup>1</sup> Označení pro technologie interaktivních webových aplikací, které mění obsah webových stránek bez nutnosti jejich znovunačtení.

**Tab. 1** Rozdíl mezi tenkým a tlustým klientem

	Místní úložiště	Místní zpracování
<b>Tenký klient</b>	Ne	Ne
<b>Tlustý klient</b>	Ano	Ano

*zdroj: vlastní archiv*

## 2.4 Interoperabilita a standardy poskytování dat

S rozvojem technologií GIS neustále roste počet aplikací pro tvorbu, editaci a prohlížení prostorových dat, a tím také narůstá počet formátů určených pro práci s těmito daty. Proto byly zavedeny standardy, které zajišťují interoperabilitu mezi jednotlivými aplikacemi.

Interoperabilita je schopnost programu komunikovat s jiným programem, zejména ve smyslu sdílení dat. Základní myšlenkou interoperability je zavedení standardů, které nejsou závislé na jednom konkrétním softwaru. Jedná se o standardizování datových formátů a struktur, o standardy pro definice výpočetních postupů a specifikace aplikačních rozhraní (Sklenička 2006). Jde tedy o vytvoření obecně uznávaného vzoru, podle kterého by se mělo postupovat při tvorbě GIS aplikací. Těmito standardy disponují zejména Open Source softwary, což by mělo zaručit jejich dlouhodobou životaschopnost. Naopak některé komerční společnosti modifikují standardy pouze pro své produkty, a tím interoperabilitu narušují. Hlavním subjektem, který se zabývá interoperabilitou v oblasti GIS, je Open Geospatial Consortium (OGC).

OGC je mezinárodní neziskové konsorcium založené roku 1994, skládající se z více než 400 komerčních, vládních, neziskových a univerzitních organizací. Cílem OGC je vývoj standardů, které zajistí interoperabilitu GIS aplikací. Tyto standardy jsou zveřejněné a volně dostupné na webových stránkách OGC (<http://www.opengeospatial.org>). Mezi členy OGC mimo jiné patří organizace ESRI, Google, US Geological Survey, NASA a další.

Obdobnou organizaci můžeme najít i v oblasti webových technologií. Jedná se o World Wide Web Consortium (W3C) založené stejně jako OGC, v roce 1994. W3C se zabývá vývojem protokolů a směrnic pro zajištění interoperability a dlouhodobého růstu webu, což má význam i pro geoinformatiku.

### 2.4.1 Základní OGC standardy pro sdílení geografických dat

V poslední době dochází k přechodu řady aplikací z desktopového prostředí na internet, do prostředí distribuované správy. To se týká i GIS, kde dochází ke vzniku internetových mapových aplikací. Tyto aplikace pak využívají webové mapové služby poskytující geodata, které mohou být zajištěny pomocí OGC standardů.



OGC standardy jsou dokumenty popisující rozhraní a šifrování služeb, kterých využívají aplikace pracující s geodaty. Pod tyto standardy spadají tzv. OGC Web Services (OWS), což jsou standardy určené pro webové aplikace. OWS služby poskytují geodata v různých formátech a podporují různé typy dotazů. Následují příklady nejpoužívanějších OWS služeb.

### A. Web Map Service (WMS)

WMS je jednou z nejvyžívanějších specifikací OGC poskytující data v rastrovém formátu. Zdrojovými daty však nemusí být nutně rastrová data. Server WMS může obsahovat jak data rastrová, tak vektorová, a až podle požadavku klienta vybere služba potřebná data, z kterých vygeneruje mapu v rastrovém formátu, kterou odešle klientovi. Klientem zde může být obyčejný webový prohlížeč.

WMS podporuje tyto dotazy:

- *GetCapabilities*: Dotaz vrací XML dokument s metadaty, popisující obsah služby - jaké vrstvy jsou dostupné, popis těchto vrstev a formát rastru, který služba vrací. Tyto informace slouží aplikacím pro další spolupráci se serverem.
- *GetMap*: Vrací mapu na základě požadavků klienta uvedených v URL.
- *GetFeatureInfo*: Nepovinný dotaz, který vrací informace v XML formátu o objektech nacházejících se na zadaných souřadnicích. Tento dotaz je podporován pouze u vrstev, které mají definovaný „queryable“ atribut.

Pokud WMS poskytuje všechny tři dotazy, jedná se o „queryable WMS“. Vedle toho „Basic WMS“ podporuje pouze první dva dotazy.

### B. Web Feature Service (WFS)

WFS, jakýsi protějšek WMS, poskytuje vektorová data ve formátu GML (Geography Markup Language), který byl vyvinut konsorciem OGC. WFS tak umožňuje práci přímo se zdrojovými daty, protože server poskytuje přímo zdrojová data (ne pouze jejich obrazovou reprezentaci jako u WMS). U tohoto standardu je vyžadována práce s tlustým klientem.

WFS podporuje tyto dotazy:

- *GetCapabilities*: Stejně jako u WMS je výsledkem dotazu XML dokument s metadaty, popisující obsah služby.
- *GetFeature*: Vrací XML soubor s požadovaným výběrem prvků.
- *DescribeFeatureType*: Vrací XML dokument, popisující typy prvků poskytované službou WFS.

- *GetGmlObject*: Vrací požadované GML objekty.
- *GetPropertyValue*: Vrací hodnotu atributu vybraného prvku/ů.
- *Transaction*: Pomocí transakční operace může klient vytvořit, modifikovat, přemístit nebo odstranit prvky z datového úložiště WFS
- *LockFeature*: Pokud klient modifikuje prvek uložený v databázi WFS, požadavek umožní jeho uzamčení, aby se zabránilo jinému klientovi modifikovat tentýž prvek ve stejný okamžik.
- *GetFeatureWithLock*: Kombinace požadavků *GetFeature* a *LockFeature*.

U WFS se rozlišují tři třídy. Pokud jsou implementovány pouze povinné dotazy *GetCapabilities*, *GetFeature* a *DescribeFeature*, jedná se o „Basic WFS“. U třídy „Xlink WFS“ je navíc implementován dotaz *GetGmlObject*. A třída „Transactional WFS“ implementuje kromě základních dotazů ještě služby *Transaction* a *LockFeature*.

### **Geography Markup Language (GML)**

Jedná se o rozšířený standard pro popis geodat umožňující jejich přenos a ukládání. Má stejnou syntaxi jako XML, pomocí kterého byl vytvořen a je platformě nezávislý-jedná se o otevřený formát. GML nepopisuje vzhled, ale strukturu geografických prvků s jejich vlastnostmi, a jeho součástí jsou také geometrické a topologické principy. Jazyk GML využívají mimo jiné společnosti ESRI či ČÚZK.

(Říha 2007)

### **C. Web Coverage Service (WCS)**

WCS poskytuje prostorová data jako „coverage“ - digitální data reprezentující určitý fenomén, který se mění s místem. Může se jednat o družicové snímky, letecké snímky nebo o digitální model terénu. WCS, na rozdíl od WMS, poskytuje data s jejich detailním popisem včetně časového údaje. Díky tomu lze pomocí této služby provádět různé druhy analýz.

WCS podporuje tyto dotazy:

- *GetCapabilities*: Vrací metadata a seznam coverage vrstev, které služba poskytuje.
- *GetCoverage*: Vrací data v požadovaném rozsahu se všemi vlastnostmi.
- *DescribeCoverage*: Vrátí XML dokument popisující data v zadaném rozsahu.

## D. Coordinate Transformation Service (CT)

CT je standard poskytující specifikace a přístup k službám, určených k transformaci prostorových dat mezi jednotlivými souřadnicovými systémy. Tato služba je vhodná při současné práci s daty v různých souřadnicových systémech. Souřadnice dat mohou mít libovolný počet dimenzí. CT tedy může operovat jak s 2D, 3D i 4D daty.

### 2.4.2 EPSG kódy a souřadnicové systémy

Aby bylo možné všem zájmovým prvkům přiřadit souřadnice jednoznačně určující bod na zemském povrchu, je třeba zavést souřadnicový systém, který je vztažen k určitému modelu Země. Systém je charakterizován referenční plochou (elipsoidem, rovinou, koulí) s jejími parametry, typem zobrazení referenční plochy na rozvinutelnou plochu, definicí počátku, souřadnicových os a jednotek míry kartézského (pravoúhlého) systému a způsobem vybudování základní trigonometrické sítě (Lenhart 2000). Souřadnicových systémů je několik, a tak na jeden bod na Zemi může odkazovat několik různých souřadnic. Proto bylo třeba zavést databázi, pomocí které se identifikuje, o jaký souřadnicový systém se jedná a jaké má tento systém parametry.

Roku 1985 společnost European Petroleum Survey Group (vědecká společnost s vazbami k evropskému ropnému průmyslu) vytvořila pro vnitřní potřebu databázi souřadnicových systémů pojmenovanou podle názvu společnosti jako „EPSG Geodetic Parameter Dataset“. V roce 2005 tato společnost zanikla a správu databáze převzala nově vzniklá asociace The International Association of Oil & Gas producers (OGP). Název databáze však zůstal ponechán, zkráceně se používá označení EPSG (nezaměňovat s názvem zaniklé společnosti).

Každý souřadnicový systém v databázi má přiřazen jedinečný kód, podle kterého je identifikován, a ke každému souřadnicovému systému jsou uvedeny jeho určující parametry. Databáze je volně dostupná na stránkách EPSG (<http://www.epsg.org/Geodetic.html>).

**Tab. 2** Významné souřadnicové systémy a jejich parametry

EPSG kód	Název souřadnicového systému	Nultý poledník	Jednotky	Význam
102067	S-JTSK Křovák East North	Greenwich	metry	uplatnění na území ČR a SR
4326	WGS 84	Greenwich	stupně	celosvětové využití, používá se např. pro GPS
3857	Spherical Mercator	Greenwich	metry	je využíván Google Maps, OpenStreetMaps a dalšími

*zdroj: vlastní archiv*

## KAPITOLA 3

### Mapové servery

Mapové servery jsou důležitou součástí GIS technologií, neboť jsou základem pro šíření prostorových dat na internetu. Hlavním úkolem mapového serveru je vykreslování digitálních map, s kterými lze provádět různé operace. Mapové servery by měly splňovat standardy, o kterých byla řeč v předchozí kapitole, a tím zajistit interoperabilitu s jinými aplikacemi. Díky těmto standardům nemusí mapový server přistupovat jen k datům z vlastního úložiště, ale i k datům uloženým na vzdálených serverech. V současnosti je k dispozici mnoho řešení pro vytvoření vlastního mapového serveru. Na trhu jsou jak komerční řešení, které zastupuje například ESRI ArcGIS Server, tak nekomerční řešení s hlavním představitelem UMN MapServerem.

Mapový server funguje na architektuře klient - server, a v podstatě se jedná o GIS systém, který se ovládá pouze pomocí parametrů - textově. Klientem se zde rozumí počítač, který posílá požadavky na vykreslení mapy (přes webový prohlížeč), a serverem jiný počítač (může však být tentýž počítač), který na tyto požadavky reaguje (Guo 2007).

### 3.1 Komponenty mapového serveru

#### Počítač

Počítač je samozřejmou součástí mapového serveru. Je ale dobré zmínit, že složitější mapové servery vyžadují výkonnější počítače. Čím je mapa komplexnější, tím zabere více času mapu vygenerovat, a je dobré tento čas minimalizovat. Nejpodporovanějším operačním systémem pro práci s mapovým serverem je Windows a Linux. Mac OS X a jiné systémy založené na Unixu jsou podporovány méně.

#### Připojení k internetu

Pro přístup klientů k mapovému serveru, je třeba zajistit připojení k internetu. I zde platí, že čím obsáhlejší data poskytujeme, tím je třeba rychlejší připojení k minimalizaci prodlevy mezi zadáním požadavku klientem a odpovědí mapového serveru.

## Webový server

Webový server obstarává komunikaci mezi samotnou mapovou službou a koncovým uživatelem. Server zobrazuje mapy a nástroje potřebné k jejich ovládání. Nejvíce se využívají produkty Apache a Microsoft Internet Information Server (IIS).

## Mapový server

Mapový server generuje mapy, které vidí koncový uživatel na webové stránce. Mapový server tedy musí komunikovat s webovým serverem a ze zdrojových dat vytvářet obrázky na základě požadavků klienta.

## Prostorová data

K vykreslení mapy jsou potřeba data obsahující prostorovou informaci. Tato data musí být ve formátu, který daný mapový server podporuje, a mohou být uložena jak na místním počítači, tak na vzdáleném serveru (k datům lze přistupovat pomocí zmíněných služeb jako je WMS, WFS atd.).

## Metadata

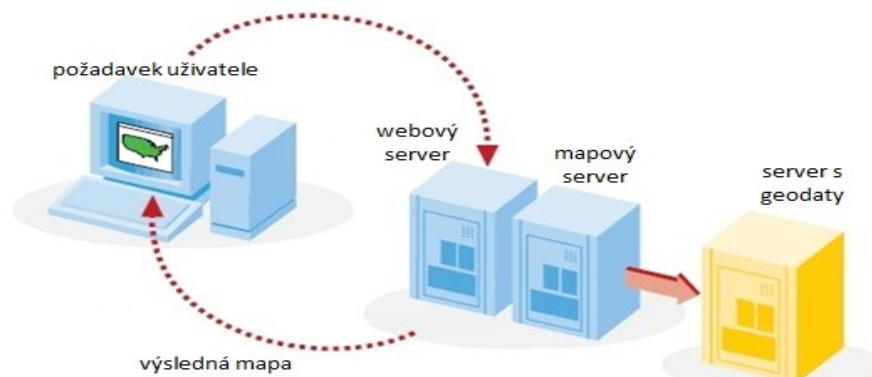
Metadata jsou data o datech. Popisují odkud data jsou, co obsahují, v jakém jsou souřadnicovém systému, kdo je pořizovatelem dat atd. Pro běh mapového serveru nejsou nezbytně nutné, ale s přibýváním dostupných dat na internetu je třeba tato data katalogizovat.

(Mitchell 2005)

## Proces přenosu informací mezi klientem a mapovým serverem:

Klient zadá požadavek přes HTTP protokol webovému serveru, který naváže spojení s CGI aplikací (mapovým serverem) - spustí ji s parametry daného požadavku. Mapový server zpracuje požadavek a vygeneruje výslednou mapu, jejíž zdrojová data mohou být uložena lokálně nebo mohou být získána ze vzdáleného serveru. Výsledná mapa je následně předána webovému serveru, který ji zařadí do HTML kódu a odešle zpět klientovi (Cajthaml 2005).

**Obr. 1** Schéma mapového serveru



*zdroj: (Mitchell 2005)*

## 3.2 Používané technologie mapových serverů

### A. ESRI ArcGIS Server

ESRI ArcGIS Server spadá do kategorie komerčních serverů a je patrně nejrozšířenější v této skupině produktů. Jedná se o komplexní nástroj poskytující správu, vizualizaci, a editaci prostorových dat. Navíc server nabízí pokročilé analytické funkce. Ke službám ArcGIS Serveru mohou přistupovat různí klienti. Kromě tradičních webových aplikací lze k serveru přistupovat z mobilních aplikací, z prostředí ArcGIS Desktop – vhodné zejména pro pokročilejší analytické funkce, nebo z ArcGIS Explorer, díky kterému není nutné instalovat nástroje GIS na vlastním počítači. Pro koncové uživatele jsou k dispozici již hotová řešení jak webových, tak mobilních aplikací, díky kterým není třeba nic programovat. Pokud však tato řešení uživatelům nevyhovují, jsou k dispozici vývojové nástroje, které usnadňují tvorbu vlastních aplikací. Server je také možné rozšířit několika volitelnými nadstavbami, mezi které patří například 3D Analyst, Network Analyst či Workflow Manager, který umožňuje řídit a organizovat pracovní postupy zaměstnanců.

### B. UMN MapServer

Stejně jako ArcGIS zaujímá přední postavení mezi komerčními mapovými servery, MapServer stojí v popředí mezi Open Source řešeními. MapServer byl vyvinut v 90. letech na univerzitě v Minnesotě ve spolupráci s NASA a Minnesota Department of Natural Resources. V současnosti je ve správě OSGeo (Open Source Geospatial Foundation) a na jeho vývoji pracuje více než 20 společností z celého světa. MapServer je nástroj pro vykreslování geodat napsaný v jazyku C, jehož hlavní výhodou je široká formátová a platformní kompatibilita. Samozřejmostí je podpora mnoha OGC standardů a formátů, používaných pro ukládání geodat (ESRI shapefile, PostGIS, ESRI ArcSDE, Oracle Spatial, MySQL a další). Díky knihovně Proj.4 si MapServer poradí s tisíci různými kartografickými projekcemi. MapServer má poměrně sofistikovaný kartografický výstup, uživatel si může plně přizpůsobit podobu výsledné mapy svým potřebám. Lze nastavit různé vykreslování dat v závislosti na měřítku, vykreslování popisků jednotlivých prvků, aniž by se popisky překrývaly, nebo automatické vykreslení mapových prvků (legenda, měřítko, referenční mapa). Dále podporuje několik oblíbených vývojářských prostředí - PHP, Python, Perl, Ruby, Java, a .NET.

### C. Ostatní

Kromě dvou popsaných řešení se nabízí celá řada jiných nástrojů pro sdílení digitálních map na internetu a to jak komerčních, tak nekomerčních. Obecně bych řekl, že komerční řešení nabízejí mnohem vyšší funkcionalitu a jsou více propracované co se týče vytváření a údržby mapové aplikace z pohledu konečného uživatele. Jak bylo řečeno, ArcGIS Server nabízí uživatelům již hotová řešení mobilních a webových aplikací a spoustu jiných nástrojů. U MapServeru si tato řešení musí uživatel obstarat sám, i když dnes už se na internetu dá najít mnoho šablon, které tuto práci

výrazně usnadní. Zkrátka nekomerční řešení nejsou tak „user-friendly“. Na druhou stranu nekomerční řešení by díky Open Source standardům měly zaručit interoperabilitu s ostatními aplikacemi, podporu mnoha formátů vstupních dat, širší možnosti přizpůsobení si programu vlastním potřebám a výsledná cena implementace je s cenou komerčních řešení nesrovnatelná. Je také třeba zmínit, že Open Source řešení nejsou plnohodnotnými GIS systémy, jako některé komerční alternativy.

Z dalších komerčních řešení stojí za zmínku Autodesk MapGuide, MapInfo MapXtreme a Intergraph Geomedia. Z nekomerčních pak GeoServer, MapGuide Open Source, GRASSLinks, Degree, ALOV Map či JShape.

### 3.3 Web Mapping API

Mapový server funguje na architektuře klient – server, kdy klient komunikuje s mapovým serverem přes webové rozhraní. Toto rozhraní si může uživatel naprogramovat sám nebo lze využít různé API, které nabízí mnoho užitečných funkcí. API (Application Programming Interface) je specifikace, která funguje jako rozhraní ke komunikaci mezi aplikacemi. Web Mapping API slouží k vytvoření webové interaktivní mapy, kterou lze umístit na jakoukoliv webovou stránku.

Vedle klasických API od Google Maps, Yahoo Maps nebo Microsoft Bing Maps, které umožňují vytvoření webových rozhraní pro prohlížení mapových vrstev daných společnostmi s různými funkcemi, můžeme najít mnoho jiných komerčních a Open Source API.

Mezi komerční patří např. BigTribe API, které mimo prohlížení map třetích stran umožňuje zobrazení vlastního obsahu a nabízí funkce typu nalezení nejbližšího bodu zájmu. DeCarta API navíc umožňuje geokódování a routing, GlobeXplorer API nabízí přístup k světově největší databázi leteckých a družicových snímků a konečně Poly9 FreeEarth API nabízí 3D pohled na Zemi, podobně jako Google Earth.

V kategorii Open Source jsou hlavními představiteli OpenLayers a MapBuilder API. OpenLayers umožňují vytvářet webové aplikace podobné Google maps, podporuje OGC standardy WMS a WFS a umožňuje zobrazovat geodata ve formátu GeoRSS, GML, KML a GeoJSON. MapBuilder má velmi podobnou funkcionalitu.

## KAPITOLA 4

### Použitá data a metodika práce

Hlavním cílem práce bylo vytvořit mapový server Albertova a okolí a to výhradně pomocí open source technologií. Pro vykreslování geodat byl zvolen UMN MapServer a pro vytvoření webového rozhraní bylo použito API OpenLayers. Na stránkách MapServeru lze najít dobře zpracovanou dokumentaci a hlavně tutoriál, který byl ideálním pomocníkem při prvním seznámení se s programem a jeho funkcemi. Mimo to lze na internetu najít mnoho fór a diskuzí zabývajících se problematikou MapServeru, které mi pomáhaly při řešení nejrůznějších komplikací.

Jako hlavní podkladové mapové vrstvy byly použity volně dostupné vrstvy OpenStreetMaps (Mapnik tileset) a Google Maps (základní a letecké), ke kterým je přistupováno jako ke vzdálené službě WMS. Vrstvy budov a ortofoto snímků byly přidány jako WMS vrstvy generované MapServerem. Vektorová vrstva budov slouží zároveň jako informační vrstva, proto bylo třeba naplnit atributovou tabulku patřičnými údaji a jednotlivé budovy vizualizovat podle typu fakulty. Informační prvek byl zajištěn požadavkem „WMSGetFeatureInfo“, který vrací informace o jednotlivých budovách.

#### 4.1 Struktura MapServeru

MapServer je ve svém základu CGI aplikace reagující na příkazy webového serveru. Primární funkce MapServeru je čtení dat z různých zdrojů, jejich zpracování a vykreslení do jednoho výsledného obrázku. Klíčovými komponenty MapServeru jsou spustitelná aplikace CGI, *Map File*, šablona a zdrojová data. Je třeba si uvědomit, že MapServer není plnohodnotným GIS systémem. Nenabízí integrovaný databázový systém, nemá nástroje pro georeferencování a jeho analytické možnosti jsou omezené (Kropla 2005).

##### Spustitelná aplikace CGI

V angličtině se také používá označení „MapServer Executable“. Je to hlavní „motor“ celého mapového serveru. CGI aplikace v podstatě nečinně „sedí“ mezi webovým serverem a zdrojovými daty. V momentu kdy klient zadá požadavek k vykreslení mapy se aplikace spustí s patřičnými



parametry, vygeneruje požadovanou mapu a opět se její běh zastaví (viz obr. 2). Tento proces se spustí URL odkazem, který může vypadat následovně:

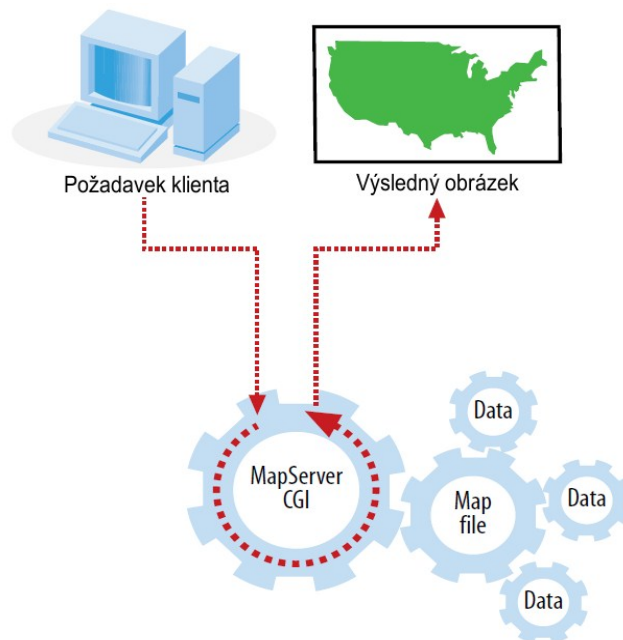
*"localhost:8080/cgi-bin/mapserv.exe?map=ms4w/Apache/htdocs/kampus.map&mode=map"*

Odkaz v první části definuje cestu ke spustitelné aplikaci MapServeru „mapserv.exe“ a následně je uvedena cesta k souboru *Map File* (zde „kampus.map“). Poslední parametr *mode* určuje, co má aplikace s vygenerovanou mapou provést. Parametr *mode* může být nastaven na hodnotu „map“, „browse“ nebo „query“.

Při spuštění MapServeru v módu „map“ je vygenerovaná mapa předána klientovi přímo a dále s mapou nelze provádět žádné operace - jedná se o statickou mapu. V módu „browse“ může klient mapu ovládat. Aplikace zaznamená pozici kliknutí, úroveň přiblížení, směr a další informace, které jsou klientem odeslány zpět MapServeru. MapServer tyto hodnoty nahradí za proměnné v HTML šabloně a na jejich základě vytvoří novou mapu, kterou vrátí klientovi.

V obou případech ale uživatel nemá přístup k atributovým ani prostorovým údajům o datech. Právě tyto informace je možné poskytnout v módu „query“, a to jak v tabulkové tak grafické podobě.

**Obr. 2** *Struktura MapServeru*



*zdroj: (Mitchell 2005)*

## Map File

*Map File* je textový soubor s příponou *.map*, pomocí kterého tvůrce mapy konfiguruje její výslednou podobu. Skládá se z několika objektů a ke každému objektu se přiřazují patřičné parametry. Základní aplikace musí obsahovat „Map“, „Web“ a „Layer“ objekty („Web“ objekt může být vynechán, pokud se aplikace spouští v módu „map“). Nejdříve se definují parametry společné pro celou mapu: název mapy, formát výstupu, rozsah mapy, výsledná projekce, metadata. Poté následuje seznam jednotlivých vrstev a jejich parametry: umístění zdrojových dat a typ jejich formátu, souřadnicový systém a nastavení grafické podoby výstupu. Dále je možné nastavit vykreslení legendy, měřítka, či popisků jednotlivých prvků.

**Obr. 3** Ukázka *Map File*

```
MAP
  NAME          "kampus"
  IMAGETYPE      PNG
  EXTENT         -743187.36 -1045307.30 -742491.40 -1044467.53
  PROJECTION     "init=epsg:900913"
END

WEB
  TEMPLATE      'kampus.html'
  IMAGEPATH     '/ms4w/tmp/ms_tmp/'
  IMAGEURL      '/ms_tmp/'
END

LAYER
  NAME          budovy
  DATA         "kampus"
  TYPE          polygon
  PROJECTION     "init=epsg:102067"
END
  STATUS        on
  CLASS
    NAME        "PřF"
    STYLE
      COLOR      0 205 0
      OUTLINECOLOR 20 20 20
    END
  END
END
END
```

*zdroj: vlastní archiv*

Pro tvorbu *Map File* postačuje obyčejný textový editor. Existují ale i nástroje, které *Map File* automaticky vygenerují. Tuto funkci umožňuje např. Open Source software QuantumGIS, disponující zásuvným modulem „MapServer Export“. Uživatel si jednoduše nakonfiguruje podobu mapy v grafickém prostředí GIS a nemusí psát žádný kód. Jak to ale bývá u automaticky

generovaných kódů, ne vždy se vygeneruje jak by si uživatel přál. Proto je ve většině případů vhodnější editovat Map File samostatně.

## Šablona

Šablona definuje podobu webového rozhraní, přes které se ovládá interaktivní mapa generovaná MapServerem. Stejně jako *Map File* je i šablona textový soubor ovšem s příponou *.html*. Šablona se skládá z konvenčních HTML tagů a CGI proměnných specifických pro MapServer. Parametry šablony určují jaké vrstvy mají být viditelné, jak má být mapa přiblížena, kam se má mapa posunout atd. Tyto parametry jsou proměnné, které se po vyřízení požadavku klienta nahradí aktuálními hodnotami.

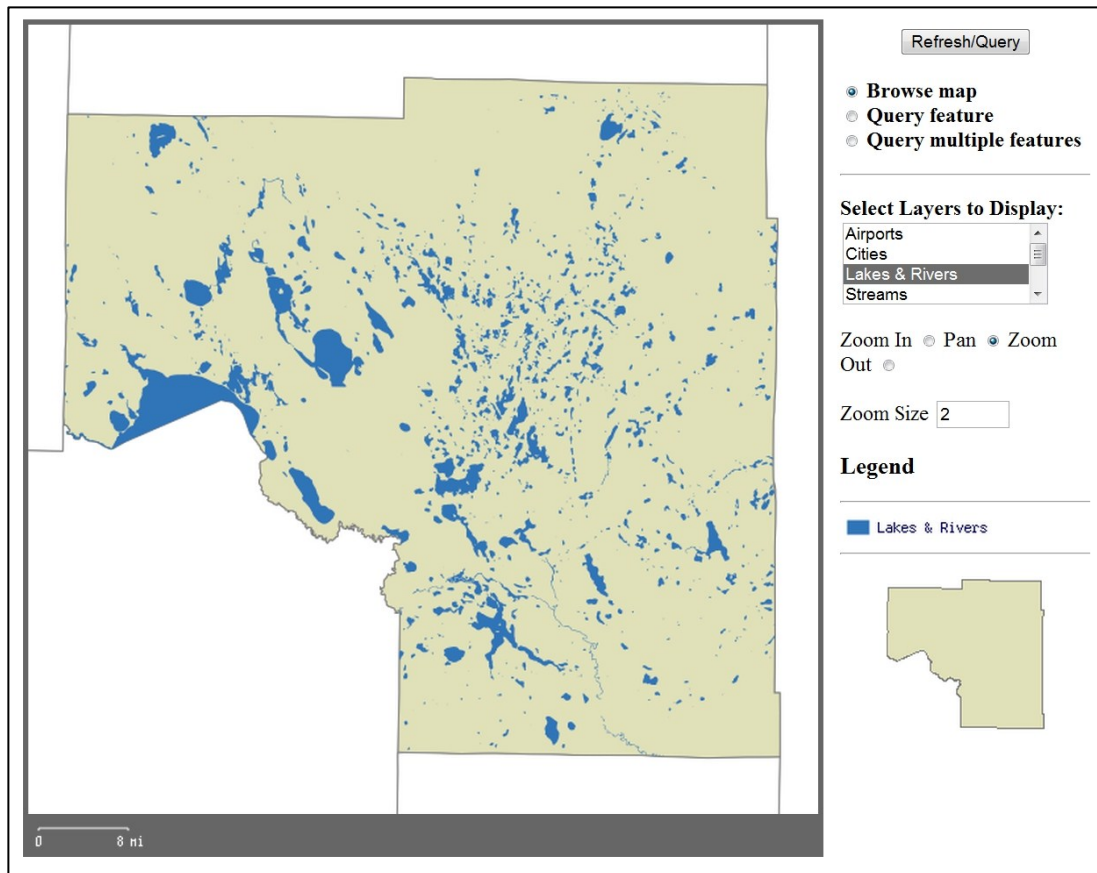
Šablony mohou být dvojího druhu - *HTML templates* a *Query templates*. *HTML templates* jsou šablony vrácené přímo klientovi - zajišťují vykreslení webového rozhraní. *Query templates* se využívají pro zobrazení výsledků prostorových dotazů. Lze tak nejen prohlížet prostorová data, ale i dozvědět se o nich patřičné informace.

Možnosti šablon napsaných pouze HTML kódem jsou limitované a jejich vzhled je velmi jednoduchý (viz obr. 4). Ovládací prvky zde tvoří jen obyčejné formuláře. Zápis kódu, který vytvoří formulář pro výběr mapové vrstvy jako je na obr. 4, vypadá následovně:

```
<b>Select Layers to Display: </b><br>
<select multiple name="layer" size=3>
  <option value="airports" [airports_select]> Airports
  <option value="cities" [cities_select]> Cities
  <option value="lakespy2" [lakespy2_select]> Lakes & Rivers
  <option value="dlgstln2" [dlgstln2_select]> Streams
</select>
```

Tučně zvýrazněné prvky představují HTML tagy, modře jsou zvýrazněné specifické proměnné MapServeru, které se odkazují na objekty definované v příslušné Map File. Část v hranatých závorkách se při běhu CGI aplikace MapServeru nahradí aktuální hodnotou.

Pro vytvoření profesionálnějšího rozhraní je možné využít JavaScript, který se zakomponuje do HTML kódu. Pokud tvůrce této „pokročilejší“ verze šablony nechce psát celý kód sám, lze na internetu najít předpřipravená řešení, nebo je možné využít vhodných „Web Mapping API“ zajišťující webové rozhraní mezi klientem a MapServerem.

**Obr. 4** Ukázka webového rozhraní definovaného šablonou „itasca\_basic.html“

zdroj: vlastní archiv

### Zdrojová data

Jednou z výhod MapServeru je podpora celé řady datových formátů. Mimo nejběžnějších formátů jako ESRI shapefile nebo Tiff/GeoTiff, které jsou MapServerem podporovány defaultně, je možné použít desítky jiných formátů za pomoci Open Source knihoven GDAL (pro rastrová data) a OGR (pro vektorová data). Pokud MapServer přistupuje k datům uloženým lokálně, mohou být jednoduše uloženy v libovolné složce, nebo lze data uložit do databáze. K tomuto účelu existuje mnoho rozšíření, která umožňují v databázi uchovávat prostorová data. Hojně využívanou databází je PostgreSQL databáze s rozšířením PostGIS. Tato databáze zvládá kromě uchovávání prostorových dat také manipulaci s daty pomocí SQL příkazů. Jinou variantou může být databáze MySQL.

## 4.2 Zprovoznění MapServeru

Možností, jak zprovoznit MapServer, je více. Vzhledem k tomu, že se jedná o Open Source software, je možné stáhnout si zdrojový kód, upravit ho dle konkrétních potřeb a zkompileovat do strojového kódu. To je ale možnost spíše pro profesionály, kteří vyžadují specifické chování

aplikace. Proto byla zvolena možnost jednodušší, bez nutnosti kompilace zdrojového kódu a to instalace pomocí balíčku „MapServer For Windows“ (MS4W). Tento balíček lze stáhnout z webu MapTools.org, kde je zveřejněno mnoho jiných Open Source projektů týkajících se digitálního mapování. MS4W je zipovaný soubor, který obsahuje vše potřebné k zprovoznění MapServeru včetně webového serveru Apache. Stažený MS4W soubor stačí extrahovat do kořenového adresáře a nainstalovat Apache.

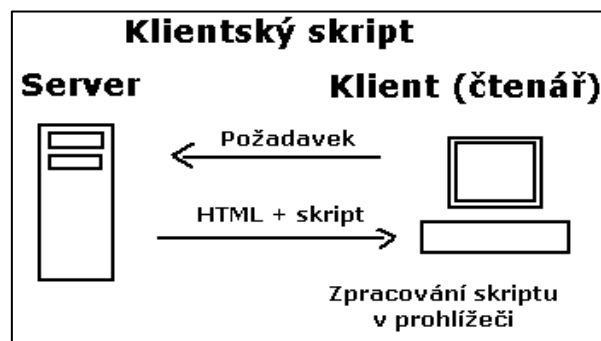
### 4.3 Tvorba webového rozhraní pomocí API OpenLayers

Webové rozhraní pro MapServer zajišťují šablony. Mohou být psány jen HTML kódem se specifickými proměnnými MapServeru, nebo se dají rozšířit JavaScriptem a tím zajistit efektnější vzhled. V mém případě byla použita JavaScriptová knihovna OpenLayers.

#### 4.3.1 JavaScript

JavaScript je interpretovaný, multiplatformní programovací jazyk se základními objektově orientovanými schopnostmi. Je určen především k oživení WWW stránek - dynamicky vytváří HTML obsah. JavaScript se zapisuje přímo do HTML kódu a odesílá se i s HTML stránkou na klienta (do prohlížeče), kde je interpretován - prohlížeč vykoná operace deklarované JavaScriptem. Tento způsob zpracování skriptu se také označuje jako klientský skript.

*Obr. 5 Schéma klientského skriptu*



zdroj: [www.jakpsatweb.cz](http://www.jakpsatweb.cz)

#### Charakteristiky a výhody jazyka:

- interpretovaný - nemusí se kompilovat
- objektový - využívá objektů prohlížeče a zabudovaných objektů
- podporuje ho většina prohlížečů
- syntaxe podobná jazykům C či Java - výhoda pro uživatele ovládající tyto jazyky

**Nevýhody jazyka:**

- JavaScript funguje pouze v prohlížeči
- uživatel může JavaScript zakázat
- existují různé odlišné verze jazyka i prohlížečů, což vede k častým chybám
- neumí přistupovat k souborům (kromě cookies) ani k žádným systémovým objektům

Jedná se tedy o druhořadý jazyk, použitelný jen v HTML stránkách (Janovský 2012).

**JavaScriptová knihovna**

Pro zrychlení a zefektivnění vývoje programů se dají použít JavaScriptové knihovny. Jedná se o soubor podpůrných komponent, zahrnující připravené třídy a metody objektů.

*Využití:*

- tvorba tlačítek a navigačních menu
- využití animačních efektů
- zpracování událostí při ovládání myši nebo z klávesnice
- ovládání zobrazení částí obsahu a grafiky

(Nymš 2012)

**4.3.2 Struktura OpenLayers**

OpenLayers jsou JavaScriptovou knihovnou, proto zde není nutná žádná instalace, jako je tomu u MapServeru. Stačí stáhnout zazipovaný balíček ze stránek OpenLayers ([www.openlayers.org](http://www.openlayers.org)) a ten extrahovat do libovolného umístění. Webové rozhraní OpenLayers se potom vytvoří pomocí HTML dokumentu, ve kterém musí být tag „script“ odkazující se na knihovnu OpenLayers. OpenLayers mají dva základní objekty: „Map“ a „Layer“.

**Map**

Tento objekt je nadřazený všem vrstvám. Nese informace o projekci, rozsahu, jednotkách a dalších parametřů mapy. Tomuto objektu se přiřazují všechny vrstvy a ovládací prvky mapy.

**Layer**

Layer - vrstva je zdrojem dat. Každé vrstvě se deklarují informace, jak má k datům přistupovat a jak se má zobrazit. Podporovány jsou jak vrstvy rastrové tak vektorové a to v několika různých formátech. Mimo klasické rastrové formáty typu Image lze přidat vrstvu Google či OpenStreetMap, poskytující data daných společností a dále vrstvy WMS a MapServer, vhodné pro mapy generované MapServerem. Mezi podporované vektorové formáty patří mimo jiné KML, GML, GeoJSON či WFS. OpenLayers mají dva základní typy vrstev: podkladové vrstvy - Base Layers a překryvné vrstvy - Overlays.

### Base Layers

Podkladové vrstvy mají tu vlastnost, že v daný okamžik může být viditelná právě jedna. Aktuálně zvolená vrstva pak určuje souřadnicový systém a dostupné úrovně přiblížení mapy. Base Layers jsou většinou rastrové vrstvy, které mívají parametr „isBaseLayer“ defaultně nastaven na hodnotu „true“.

### Overlays

Na rozdíl od podkladových vrstev může být najednou aktivováno několik překryvných vrstev overlays. Tyto vrstvy nemají vliv na dostupné úrovně přiblížení mapy, ale mohou být nastaveny tak, aby se v závislosti na aktuálním přiblížení aktivovaly nebo deaktivovaly.

Pro vytvoření webového rozhraní pomocí OpenLayers ve své nejjednodušší podobě stačí do základního kódu jednoduchými příkazy přidat mapové vrstvy různých typů (viz obr. 6). Toto základní rozhraní umožňuje jen zoomování a posouvání mapy (viz obr. 7). Pro přidání dalších prvků, jako panelu pro přepínání mapových vrstev, nástroje pro přiblížení na vybranou oblast nebo měřítka, je třeba využít třídu „Control“. Jedná se o sadu prvků, které ovlivňují stav mapy, nebo zobrazují dodatečné informace.

**Obr. 6** Ukázka kódu, který do webového rozhraní OpenLayers přidá vrstvu OSM

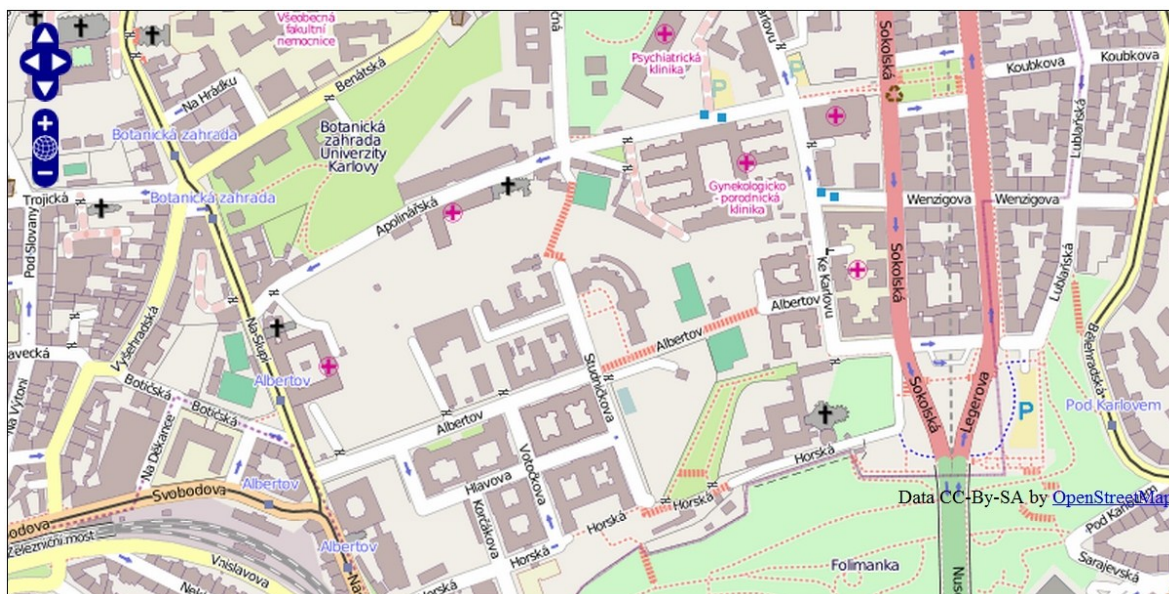
```
<html>
  <head>
    <title>OpenLayers Example</title>
    <script src="../../lib/OpenLayers.js"></script> <!--cesta ke knihovně OL-->
  </head>
  <body>
    <!--v následujícím elementu se zobrazí mapa-->
    <div id="mapElement" style="width:800px; height:400px;
      border-width:1px; border-style:solid; border-color:black;">

    <script defer="defer" type="text/javascript"> //zde začíná JavaScript
      var map = new OpenLayers.Map('mapElement'); //vytvoření objektu mapy
      var osm = new OpenLayers.Layer.OSM("OSM"); //vytvoření vrstvy OSM
      map.addLayer(osm); //přidání vrstvy osm do objektu mapy
      map.zoomToMaxExtent(); //příkaz pro načtení mapy v maximálním rozsahu
    </script>

  </body>
</html>
```

*zdroj: vlastní archiv*



**Obr. 7** Základní webové rozhraní OpenLayers s vrstvou OSM*zdroj: vlastní archiv*

## 4.4 Mapový server kampusu Albertov

### 4.4.1 Použité mapové vrstvy

Jako hlavní podkladové vrstvy byly zvoleny OpenStreetMaps a Google Maps (satelitní i základní), které jsou podporovány OpenLayers - přidají se jako vrstvy „OSM“ a „Google“. Jejich výhodou je, že se jedná o vrstvy vytvořené z dlaždic (tiles) o rozměru 256 x 256 pixelů, které vytvářejí ucelenou mozaiku mapy. Díky tomu se vždy načte jen určité území, které klient požaduje a nemusí se načítat i oblast mimo zobrazený výřez. Navíc OpenStreetMaps i Google Maps dlaždice jsou na zdrojových serverech generovány dopředu a ne až v reálném čase. Klient tak nemusí čekat na vygenerování mapových dlaždic jako je tomu u klasické WMS a prohlížení těchto vrstev je tak velmi rychlé. Obě vrstvy jsou v souřadnicovém systému Spherical Mercator, proto i všechny ostatní vrstvy musejí být převedeny do tohoto systému.

Další podkladovou vrstvou jsou ortofoto snímky ve formátu *jpg*, ke kterým je přístupováno lokálně (snímky poskytl Útvar pro rozvoj hlavního města Prahy (URM)). Přidání této vrstvy do OpenLayers bylo trochu komplikovanější. Pokud by se ortofoto snímky přidaly jako vrstva Image, což je třída OpenLayers umožňující přidat *jpg* obrázky, musel by se pokaždé načíst celý obrázek a to i v případě přiblížení jen na malou část snímku. Takové prohlížení mapy by bylo velmi pomalé. Možným řešením, které by prohlížení urychlilo, bylo použít nástroj pro vytvoření dlaždic odpovídající dlaždicím Google Maps či OpenStreetMaps. Takto vytvořené dlaždice by potom šlo načíst jako OpenLayers vrstvu „Tile.Image“ a prohlížení by se urychlilo. Znamenalo by to však práci s dalším nástrojem a možné komplikace při vytváření dlaždic pro jednotlivé úrovně přiblížení.



Proto bylo zvoleno jiné řešení. Snímky byly přidány jako WMS vrstva načítaná po jednotlivých dlaždicích. Toho lze dosáhnout pomocí MapServeru, který může fungovat jako WMS služba poskytující dlaždice odpovídající OpenStreetMaps a Google Maps. U tohoto řešení platí zmíněná nevýhoda nutnosti generování dlaždic při prvním načtení vrstvy. Jakmile se však vrstva jednou načte, je uložena do vyrovnávací paměti klienta (tzv. cache) a při následném požadavku na vykreslení dříve zobrazené části mapy se použijí tato lokálně uložená data a není nutné dlaždice znovu generovat.

Poslední a asi nejdůležitější vrstvou je vrstva „Kampus“. Jde o vrstvu budov typu overlays, která je ve formátu *shapefile* a je k ní přístupováno lokálně (data poskytl URM). Právě tato vrstva je tím hlavním přínosem celého serveru. Mimo zobrazení obrysů budov přes podkladové vrstvy poskytuje tato vrstva prostřednictvím služby „WMSGetFeatureInfo“ informace o jednotlivých budovách. I u této vrstvy se nabízelo více možností jak ji přidat do OpenLayers. Asi nejefektivnějším řešením by bylo přidat vrstvu přímo jako OpenLayers vrstvu „Vector“. OpenLayers ale neumí pracovat s *shapefile* - vrstvu by bylo třeba nejprve zkonvertovat do vektorového formátu podporovaného OpenLayers (např. KML či GeoJSON). Vzhledem k tomu, že bylo také požadováno barevně rozlišit budovy podle fakulty, zvolil jsem řešení stejné jako v případě ortofoto snímků - přidat vrstvu jako WMS generovanou MapServerem. Tento způsob je ve výsledku o něco pomalejší, ale výhodou MapServeru jsou jeho vykreslovací schopnosti - lze velmi dobře nakonfigurovat podobu výsledné mapy, případně je možné přidat popisky budov nebo automaticky generovanou legendu.

Legenda generovaná MapServerem se konfiguruje v *Map File* dané vrstvy. Prvky legendy se vytvářejí na základě jednotlivých tříd - „Class“, definovaných v *Map File*, které určují výsledný vzhled mapy (každá třída je zahrnuta do legendy). Legenda se vykresluje dohromady s mapou a nelze k ní přistupovat jako k samostatnému objektu. Proto byl pro legendu vytvořen nový *Map File*, který je v podstatě kopií *Map File* vrstvy Kampus - prvky legendy tak odpovídají prvkům mapy. V *Map File* legendy pak bylo třeba změnit některé parametry, aby se spolu s legendou nevykreslovala i její přidružená mapa. Legenda nakonec byla do OpenLayers přidána jako vrstva „MapServer“. Pomocí JavaScriptu pak lze nastavit, aby se legenda vypnula/zapnula v závislosti na viditelnosti vrstvy Kampus (obslužením události „changelayer“).

#### 4.4.2 MapServer jako WMS služba

Konfigurace WMS vrstev poskytovaných MapServerem se zajišťuje prostřednictvím *Map File*, kde musí být navíc přidán objekt „Metadata“. Metadata platná pro celý *Map File* se uvádějí v objektu „WEB“ a metadata platná pro konkrétní vrstvu v objektu „Layer“.

U objektu „Layer“ je nutné uvést pouze název vrstvy. Pro objekt „WEB“ se uvádí název WMS, HTTP odkaz na WMS, typy požadavků, které WMS zprostředkovává, EPSG kód souřadnicového

systému a poměrně důležitý parametr „ows\_http\_max\_age“, který však není povinný. Hodnota tohoto parametru se zadává v sekundách a určuje dobu, během které se nebude konkrétní požadavek na vykreslení mapy vykonávat. Pokud není tento parametr uveden, bude se požadavek klienta zpracovávat pokaždé, i když už stejný požadavek zpracován byl. Nevyužije se tak vyrovnávací paměť na straně klienta, která uchovává načtená data. Jednou zobrazené území by se vykreslovalo na straně MapServeru znovu a znovu by se vykreslená mapa odesílala klientovi. Toto chování může být užitečné u dynamicky se měnících dat, ne však u dat statických, jako v případě vrstev serveru kampusu Albertov. Proto byl tento parametr nastaven na relativně vysokou hodnotu.

#### 4.4.3 Nastavení souřadnicového systému

Při tvorbě mapového serveru, skládajícího se z několika mapových vrstev v různých souřadnicových systémech, je třeba sjednotit souřadnicový systém všech vrstev. Společný souřadnicový systém je dán souřadnicovým systémem podkladových vrstev, kterým je v případě serveru kampusu Albertov „Spherical Mercator“ (podkladové vrstvy OpenStreetMaps i Google Maps jsou poskytovány v tomto systému). Nastavení souřadnicového systému bylo třeba provést jak na straně MapServeru, tak na straně OpenLayers.

Pro vrstvy MapServeru se souřadnicový systém definuje v *Map File*. Je třeba uvést jak cílový systém (společný pro všechny vrstvy uvnitř *Map File*), tak systém zdrojových dat (pro každou vrstvu zvlášť). MapServer využívá rozšířenou knihovnu Proj.4, proto se definice projekce může skládat z několika klíčových slov nebo z EPSG kódu. V případě použití EPSG kódu, musí být uveden stejný kód včetně definice daného souřadnicového systému v textovém souboru „epsg“ (jedná se o seznam EPSG kódů a jejich definic). Definice souřadnicových systémů lze najít např. na stránce <http://spatialreference.org/ref/epsg/>. Obecně je lepší použít přímou definici souřadnicového systému než EPSG kód. Pokud totiž epsg soubor obsahuje velké množství údajů, trvá delší dobu, než MapServer konkrétní EPSG kód najde.

V případě API OpenLayers je systém definování souřadnicového systému podobný. Uvnitř objektu „map“ se definuje výsledný souřadnicový systém společný pro všechny vrstvy a uvnitř každé vrstvy se definuje projekce zdrojových dat.

#### 4.4.4 Ovládací prvky webového rozhraní OpenLayers

Základní webové rozhraní OpenLayers umožňuje jen posun mapy tažením myši a přiblížení/oddálení pohledu scrolováním nebo tlačítky defaultního ovládacího panelu. Proto byly přidány další ovládací prvky pomocí třídy „Control“, která umožňuje přidat prvky do objektu mapy samostatně nebo jako skupinu v rámci ovládacího panelu. Vzhled většiny prvků je možné upravovat pomocí jazyka CSS a jejich funkcionalitu lze rozšířit obsluhou událostí - „events“.

## A. Samostatné ovládací prvky

- **Přepínání vrstev** - „LayerSwitcher“ - přidá „rozkliknutelné“ okno s odděleným seznamem podkladových a překryvných vrstev, které lze přepnout respektive vypnout/zapnout pomocí checkboxů.
- **Měřítko** - „ScaleLine“ - parametry konstruktoru měřítka lze nastavit jednotky a maximální šířku. Pro mapy v souřadnicovém systému „Spherical Mercator“ je nutné nastavit parametr „geodesic“ na hodnotu „true“.

## B. Prvky přidané v rámci ovládacího panelu

Ovládací panel je sestava prvků, v jejímž rámci může být aktivní vždy pouze jeden prvek - aktivováním jednoho prvku se deaktivuje jiný prvek. Tyto prvky jsou ovládány prostřednictvím tlačítek - každé tlačítko má většinou definovanou podobu pro aktivní a neaktivní stav (pomocí CSS).

- **Přiblížení na vybranou oblast** - „ZoomBox“ - umožňuje tažením myši vybrat oblast, na kterou se mapa přiblíží. U tohoto prvku bylo provedeno nastavení, aby se po přiblížení automaticky aktivoval prvek pro posun mapy (obsluhou události „zoomEnd“).
- **Posun mapy** - „Navigation“ - tažením myši lze posouvat mapu. Tento prvek je v rámci panelu nastaven jako defaultní - je aktivován při prvním načtení mapy.
- **Informační tlačítko** - kliknutím na budovu z vrstvy „Kampus“ se zobrazí informace o dané budově. Prvek je přidán jako „Button“ a získání informací o budovách je zajištěno funkcí „WMSGetFeatureInfo“. Pro vrstvu Kampus musely být definovány „Query Templates“, které definují vzhled výsledku dotazu a atributy, které se mají zobrazit.
- **Nastavení středu mapy** - opět se jedná o prvek typu „Button“, který vrátí mapu na původní pozici v původním přiblížení. Toho je dosaženo funkcí „setCenter“, jejíž parametry jsou souřadnice - v jednotkách a v souřadnicovém systému jako je objekt „map“ a úroveň přiblížení - jedná se o číslo úrovně, která je dostupná pro danou mapu.

## KAPITOLA 5

### Výsledky

Výsledkem práce je interaktivní webová aplikace, která umožňuje prohlížet několik mapových vrstev Albertova a okolí, a poskytuje informace o budovách kampusu Albertov. Aplikace byla vytvořena pomocí Open Source technologií MapServer a OpenLayers.

OpenLayers zajišťují webové rozhraní, pomocí kterého se aplikace ovládá. MapServer potom slouží k vykreslení mapových vrstev uložených lokálně. Uživatel má možnost přepínat mezi několika mapovými podklady, zapnout/vypnout překryvnou vrstvu budov kampusu Albertov (dále Kampus) a volit mezi jednotlivými ovládacími prvky z panelu nástrojů. Jako hlavní podkladové vrstvy byly použity volně dostupné OpenStreetMaps a Google Maps, ke kterým je přístupováno prostřednictvím služby WMS. Další podkladové vrstvy tvoří historické ortofoto snímky poskytnuté Útvarem rozvoje hlavního města Prahy (URM). Poslední a nejdůležitější mapovou vrstvou je zmíněná vrstva Kampus poskytující informace o jednotlivých budovách. Tato vrstva také pochází z URM a byla doplněna o patřičné atributy, které si uživatel může zobrazit po kliknutí na konkrétní budovu (viz příloha 2). Jednotlivé budovy byly navíc barevně rozlišeny podle fakulty a s vrstvou je společně zobrazena legenda.

Do panelu nástrojů byla přidána tlačítka, pomocí kterých se aktivují funkce pro posun mapy, přiblížení na vybranou oblast, nastavení středu mapy na původní pozici v původním přiblížení a informační tlačítko, které umožňuje zobrazit atributové informace vrstvy Kampus.

## KAPITOLA 6

### Diskuse a závěr

Práce se zabývá širokým tématem informačních technologií pro sdílení a vizualizaci digitálních map na internetu. Rešeršní část se skládá z několika kapitol, které se snaží čtenáři toto téma přiblížit, nikoliv však podat vyčerpávající výklad. Byly probrány základní typy internetových map, webové mapové služby spadající pod standardy OGC a s tím související interoperabilita. Nakonec bylo pojednáno o mapových serverech a jejich architektuře. V praktické části jsou potom popsány nástroje, s kterými bylo pracováno, a metodika tvorby mapového serveru kampusu Albertov.

Hlavním cílem práce bylo vytvořit mapový server poskytující informace o budovách kampusu Albertov s výhradním použitím Open Source technologií. Mapový server se podařilo vytvořit, splňuje kritéria Open Source technologií, umožňuje prohlížet několik mapových vrstev Albertova a okolí, a poskytuje informace o budovách kampusu Albertov. Právě tento informační prvek je hlavním přínosem práce. Mapy Albertova a okolí jsou dostupné na mnoha jiných mapových serverech, ale neposkytují žádné další informace o zobrazených objektech.

Dostupnost vytvořené mapové aplikace na internetu je úkol k dalšímu řešení. Ideální by bylo umožnit přístup k mapové aplikaci prostřednictvím fakultního webu. V tom případě by bylo potřeba implementovat MapServer na fakultní server a buď vytvořit samostatnou webovou stránku, kam by se aplikace umístila, nebo umístit aplikaci přímo do webových stránek fakulty.

Kromě zpřístupnění vytvořené aplikace na internetu se také nabízí několik možností, jak mapový server vylepšit. Protože podklady budov Albertova a ortofoto snímky jsou statické, není je nutné při každém načtení znovu vykreslovat ze zdrojových dat (respektive z nich generovat dlaždice odpovídající Google Maps dlaždicím). Tento proces zbytečně zatěžuje server. Bylo by lepší tyto dlaždice vygenerovat dopředu, uložit je na serveru a přistupovat k nim přímo z OpenLayers. Dalším vylepšením by mohlo být umožnění zadávat atributové dotazy. Zatím aplikace umí zpracovávat prostorové dotazy, kdy se po kliknutí na nějakou budovu zobrazí její atributy. Užitečné by bylo, kdyby to šlo i obráceně - uživatel by zadal například název sekce a na základě tohoto požadavku by se zvýraznily patřičné budovy.

Doufám, že tato práce dále poslouží zájemcům, kteří by se tímto tématem chtěli zabývat.

## SEZNAM ZDROJŮ A INFORMACÍ

- CAJTHAML, J. (2005): *Využití webových mapových serverů*. In: 16. kartografická konference: Mapa v informační společnosti - abstrakta, s. 34 (plný text ve sborníku na CD), Brno 2005, ISBN 80-7231-015-1. Dostupné z: <<http://projekty.geolab.cz/gacr/a/files/cajthaml.pdf>>.
- ČERBA, O. (2008): *Digitální mapy: Přednáška z předmětu KMA/TKA* [online]. Západočeská univerzita v Plzni. 2008 [cit. 2012-05-23]. Dostupné z: <[http://gis.zcu.cz/studium/tka/Slides/digitalni\\_mapy.pdf](http://gis.zcu.cz/studium/tka/Slides/digitalni_mapy.pdf)>
- GUO, L. (2007): *Web map server* [online]. 2007 [cit. 2012-04-24]. Dostupné z: <[http://www-users.cs.umn.edu/~jkang/8715/HW7/E5\\_WebMapServer.pdf](http://www-users.cs.umn.edu/~jkang/8715/HW7/E5_WebMapServer.pdf)>.
- HETTLER, J. (2009): *Geoinformační systém Albertova a okolí*. Praha: UK Přírodovědecká fakulta. Katedra aplikované geoinformatiky a kartografie, 2009. 51 s., Vedoucí bakalářské práce Mgr. Michal Schneider.
- JANOVSKÝ, D. (2012). *Jak psát web, o tvorbě internetových stránek* [online]. 2012 [cit. 2012-05-10]. Dostupné z: <<http://www.jakpsatweb.cz>>.
- KROPLA, Bill. (2005): *Beginning MapServer: Open Source GIS Development*. Springer-Verlag, New York, 417 s.
- LENHART, Z. (2000): *Souřadnicové systémy* [online]. 2000 [cit. 2012-04-27]. Dostupné z: <<http://tvorbamap.shocart.cz/kartografie/systemy.html>>.
- MAPSERVER. (2012): *Mapserver, Open Source web mapping* [online]. 2012 [cit. 2012-05-16]. Dostupné z: <<http://mapserver.org/about.html>>.

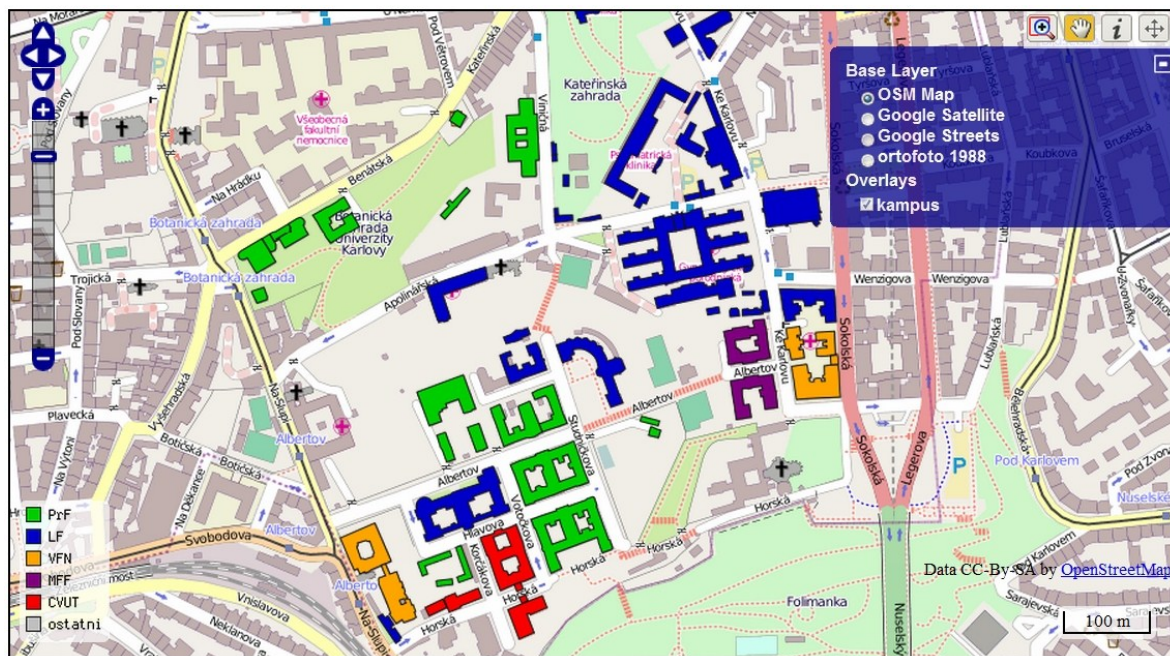
- MINÁŘ, M. (2008): *Mapová aplikace pro zobrazování epidemiologických dat na webu* [online]. Masarykova Univerzita v Brně. Fakulta informatiky, 2008. [cit. 2012-05-24]. Vedoucí bakalářské práce prof. RNDr. Jiří Hřebíček, CSc. Dostupné z: <[http://is.muni.cz/th/143200/-fi\\_b/xminar1-bc\\_prace.pdf](http://is.muni.cz/th/143200/-fi_b/xminar1-bc_prace.pdf)>.
- MITCHELL, T. (2005): *Web Mapping Illustrated*. O'Reilly Media, Sebastopol, 349 s.
- NYMŠ, J. (2012): *Informace a informatika - cesta ke hvězdám* [online]. 2012 [cit. 2012-05-10]. Dostupné z: <<http://info.spsnome.cz>>.
- OGC. (2012): *OGC® Standards and Supporting Documents* [online]. 2012 [cit. 2012-05-12]. Dostupné z: <<http://www.opengeospatial.org/standards>>.
- OGP. (2011): *Using the EPSG Geodetic Parameter Dataset* [online]. 2011 [cit. 2012-04-25]. Dostupné z: <<http://www.epsg.org/guides/docs/G7-1.pdf>>.
- OPENLAYERS. (2012): *OpenLayers: Free Maps for the Web* [online]. 2012 [cit. 2012-05-17]. Dostupné z: <<http://www.openlayers.org>>.
- ŘÍHA, J. (2007): *Distribuce map pomocí webových služeb*. [s.l.], 2007. ČVUT v Praze. Vedoucí bakalářské práce Ing. Petr Soukup, Ph.D. Dostupné z: <<http://geo2.fsv.cvut.cz/~soukup/-bkl/riha/data/!WS.htm>>.
- SKLENÍČKA, R. (2006a): *Specifikace Web Coordinate Transformation Service jako součást distribuovaných GIS* [online]. 2006 [cit. 2012-05-22]. Dostupné z: <[http://www.fce.vutbr.cz/-veda/JUNIORSTAV2007/Sekce\\_6.1/Sklenicka\\_Radek\\_CL.pdf](http://www.fce.vutbr.cz/-veda/JUNIORSTAV2007/Sekce_6.1/Sklenicka_Radek_CL.pdf)>.
- SKLENÍČKA, R. (2006): *Interoperabilita v GIS podle specifikací OGC* [online] 2006 [cit. 2012-03-22]. Dostupné z: <<http://maps.fsv.cvut.cz/~sklena/sklena/interoperabilitaOGC.pdf>>.

## **SEZNAM PŘÍLOH**

- Příloha 1    Webové rozhraní OpenLayers s vlastními ovládacími prvky
- Příloha 2    Informační prvek mapového serveru
- Příloha 3    CD s elektronickou verzí práce

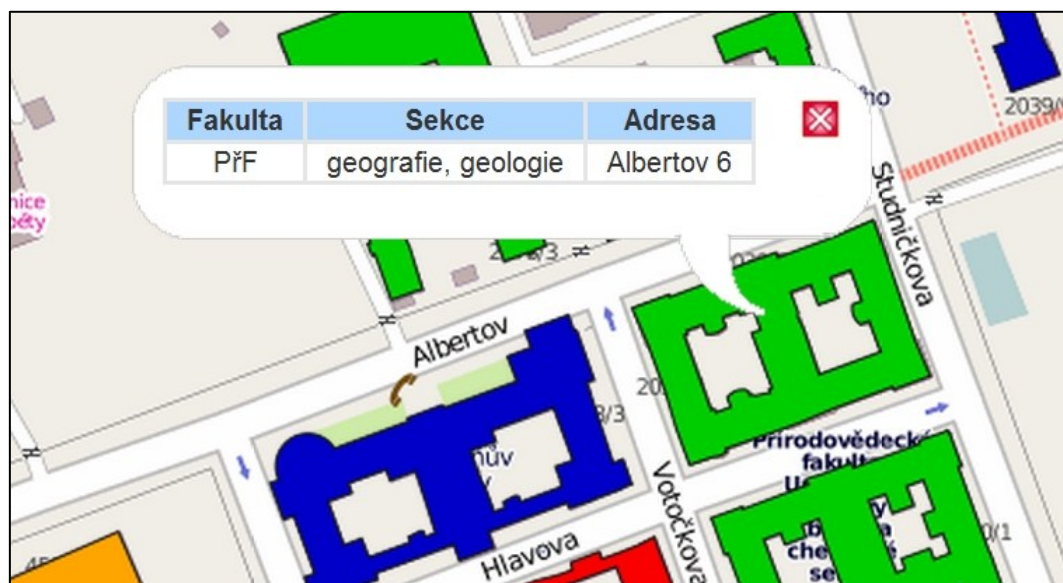


### Příloha 1 Webové rozhraní OpenLayers s vlastními ovládacími prvky



zdroj: vlastní archiv

### Příloha 2 Informační prvek mapového serveru



zdroj: vlastní archiv